

In the Matter of )  
 )  
Stanford FCC hearing ) WC Docket No. 07-52  
 )  
 )

Comments on inaccurate testimony at the FCC broadband Industry Practices Hearing  
Stanford University  
April 28, 2008

Page 1

The FCC hearing at Stanford University on April 17th 2008 was filled with inaccurate testimony from various witnesses. Since that testimony seems to be carrying significant weight both on Capitol Hill and in the media, I feel compelled to set the record straight.

### **Problems with Jon Peha's testimony**

Jon Peha testified that BitTorrent was like a telephone and implied that if a TCP reset is used by Comcast to stop a TCP stream, then that constituted a blockage of BitTorrent. Furthermore, Professor Peha implied through his telephone analogy that if BitTorrent is blocked, then the user must manually redial to reestablish the connection. These assertions are highly inaccurate and here's why.

The first problem is that Jon Peha did not understand the multi-stream aspect of BitTorrent or P2P. Peha seemed very surprised immediately before our panel at Stanford when I told him that a P2P download typically used 10 to 30 TCP streams at the same time. His surprised reply to me was "all active?" and I replied yes. The reality is that if a certain percentage of BitTorrent TCP streams are reset and temporarily blocked by an ISP, say 15% for example<sup>1</sup>, then the "Torrent" (the file that's being exchanged amongst multiple peers over the BitTorrent protocol) is essentially slowed down by an average of 15%. In other words, the "Torrent" would suffer a 15% partial blockage which is accurately described as a "delay" since the file transfer didn't actually stop. This would be like filling up a bath tub with 20 faucets and you closed 3 of those faucets. The rate of water flowing in to the tub would slow but not stop.

The second problem with Jon Peha's testimony is his implication that the user must take some sort of action to resume the BitTorrent connection or else the connection wouldn't resume. Peha's assertion can easily be proven false by a simple experiment with BitTorrent. One can easily confirm that BitTorrent will always resume a lost connection within a matter of seconds without any user intervention just by physically disconnecting a network cable on the test machine and reconnecting it. Not only does BitTorrent automatically resume, it picks up where it left off and does not need to start all over again. So even if all TCP streams in a Torrent were simultaneously blocked for a short period of time, it will quickly resume by itself and eventually finish the file transfer. Therefore this is by definition a "delay" and not a "blockage".

This is not to say that Comcast's existing form of network management is without problem because it is clear that the system has flaws and unintended consequences like the accidental blockage of IBM Lotus Notes. The use of TCP resets also have a more drastic effect on rare Torrents, which are BitTorrent files that are not popular and have few seeders or other peers who have parts of the file to download from. These rare Torrents aren't healthy to begin with and a TCP reset can in some cases trigger a complete temporary blockage. The rare Torrent will still get there eventually but it will suffer significantly more than a normal Torrent that is resilient to partial blockage.

It should be noted that BitTorrent in general is not an appropriate file transfer protocol for rare Torrents. BitTorrent tracker sites tend to rank and sort a Torrent based on the Torrent's "health" which is based on the number of available seeders and pre-seed peers. Users generally tend to avoid the "unhealthy" Torrents on the bottom of the list. Since Comcast offers a vastly superior alternative where they provide 1 gigabyte of web storage space, Comcast customers can

use that service to distribute files 10 to 20 times faster than any single Comcast BitTorrent seeder could ever provide. To further illustrate this point, Richard Bennett [posted a copy of the King James Bible](#) on his Comcast-provided web space. By contrast, I couldn't find a non-copyrighted version of the King James Bible or any non-copyrighted Barbershop music on any BitTorrent tracker site which illustrates how unlikely it is that you'll find rare or legal content on BitTorrent.

**Problems with Robert Topolski's testimony**

Robert Topolski also had problems in his testimony. Topolski, a software tester who does not work in the networking field, insists that the TCP reset mechanism isn't common in network devices and declared that I was wrong in my testimony. In my experience as a Network Engineer who designed and built networks for Fortune 100 companies, it is my experience that the TCP reset mechanism is common in routers and firewalls. For many years, Internet service providers, including LARIAT (owned and operated by Brett Glass, who has filed comments in this docket) have used TCP RST packets to protect the privacy of dialup Internet users. When a dialup user's call is over, RST packets are sent to terminate any remaining connections. This prevents a subsequent caller from receiving information -- some of which might be confidential -- that was intended for the caller who disconnected. Thus, the transmission of RST packets by a device other than the one(s) which established a connection -- for the purpose of informing the endpoints that the connection has been terminated -- is not only commonplace but salutary. Network architect Richard Bennett who works for a Router maker explained to me that TCP resets are the standard mechanism used by consumer routers to deal with NAT table overflow, which itself is typically caused by excessive P2P connections. Are we to believe a single software tester or three networking experts?

The second key problem with Topolski's testimony is that to my knowledge, he has never provided any forensic data from Comcast in the form of packet captures that can be independently analyzed. Even if Topolski did produce packet captures and we assumed that those packet captures are authentic, one man's packet captures wouldn't be a large enough sample to draw any conclusions by any legal or scientific standards. The Vuze data may constitute a large enough sample but the data isn't very granular because it doesn't tell us what percentage of reset TCP sessions are due to an ISP versus other possible sources<sup>1</sup>. The Vuze data, which even Vuze admits isn't conclusive, is highly suspect because it even shows as much as 14% resets on 10-minute TCP sessions from ISPs who do NOT use TCP reset packets to manage their network.

Furthermore, even if a TCP reset was used by Comcast at 1:45AM, we cannot assume that there was no spike in congestion at 1:45AM. As I have indicated in the past, just 26 fulltime BitTorrent seeders in a neighborhood of 200 to 400 users can consume all of the available upstream capacity in a DOCSIS 1.1 Cable Broadband network. That means less than 10% of the population seeding all day and night can cause congestion at any time of the day. Based on what little evidence presented by Robb Topolski, no conclusions can be drawn regarding the question of whether Comcast uses TCP resets for purposes other than congestion management.

---

1. The reason I use 15% as the example is because the Vuze data gathered via thousands of user's computers indicated that a Comcast broadband network typically suffered 14% to 23% for all TCP streams during 10-minute sampling periods. That 23% figure is not restricted to just BitTorrent or P2P traffic and even those TCP resets pertaining to BitTorrent aren't necessarily from Comcast. It's quite possible that the reset actually came from the client on the other end or it may have come from a customer-premise router on either end-point trying to conserve on NAT (Network Address Translation) resources. It is undeniable that a certain percentage of the TCP

reset have nothing to do with Comcast's network management practices. We really can't know what percentage of those TCP resets were due to Comcast and figuring out the exact percentage isn't trivial because there are so many factors to consider.

**Note:** Any assertions on behalf of Brett Glass and Richard Bennett that I have made in this document have been approved by Brett Glass and Richard Bennett.